

**Amendments to the Claims:**

This listing of claims will replace all prior versions, and listings, of claims in the application:

**Listing of Claims:**

1. (Currently amended) A system comprising:

a plurality of server nodes communicatively coupled on a network to serve applications over the network to a plurality of clients the plurality of server nodes organized as instances, each instance including at least one server process;

a data object to store a hierarchical representation of configuration data associated with management of the server nodes, the data object accessible by all of the servers and the hierarchy having a root, a first subset of nodes branching from the root containing configuration data associated with all servers in nodes branching from the root, a second subset of nodes branching from the root containing configuration data specific to all servers of a first instance and a third subset of nodes branching from the root of the second subset, the third subset of nodes containing configuration data specific to a first server of the first instance and a plurality of nodes branching from the root;

a property sheet logically positioned at one of the nodes of the data object, the property sheet data structure including a plurality of property names, wherein each respective property name included in the property sheet data structure is associated with a default configuration value and, optionally, a custom configuration value to pair the default configuration value to the custom configuration value for a configuration parameter represented in the data object, wherein the property sheet data structure preserves both the value of the default parameter and the custom configuration value; and

a configuration manager on at least one of the server nodes to determine whether the configuration data stored on the at least one server node is out-of-date based on the location of an updated configuration parameter within the hierarchy, wherein updated configuration parameters

within the first subset of nodes renders all servers out-of-date, updated configuration parameters within the second subset of nodes renders all servers of the first instance out-of-date and updated configuration parameters within the third subset of nodes renders only the first server out-of-date.

2. (Previously presented) The system as in claim 1 wherein the data object is stored within a central database accessible by each of the server nodes and a first node of the data object contains global configuration data associated with the plurality of server nodes and a second node of the data object contains configuration data specific to a one of the plurality of server nodes.

3. (Previously presented) The system as in claim 1 wherein, once the selected default configuration value has been modified, the default configuration value is stored independently with respect to the custom configuration values in the property sheet data structure.

4. (Previously presented) The system of claim 3, wherein the default configuration values associated with the property sheet data structure are modifiable using an interface other than a user interface.

5. (Previously presented) The system of claim 1, wherein the property sheet data structure is associated with a particular component or a set of components contained within a clustered system.

6. (Previously presented) The system of claim 1, wherein the property datasheet is modifiable with a user interface comprising:

a first dialog box to display contents of the property sheet data structure, the first dialog box including a plurality of entry rows, each respective entry row of the first dialog box including a first column to display names of corresponding properties, a second column to display current configuration values associated with corresponding properties and a third column to indicate if a configuration value displayed in the second column is a default configuration value or a custom configuration value; and

a second dialog box including a data entry field to enable a user to modify a selected default or custom value.

7. (Previously presented) The system of claim 6, wherein a custom configuration value associated with a property is modifiable by selecting the second dialog box of the corresponding property and entering a new value in the data entry field of the second dialog box.

8. (Previously presented) The system of claim 7, wherein the second dialog box of the corresponding property is selected by clicking a custom check box inside the third column of a corresponding entry row.

9. (Previously presented) The system of claim 8, wherein the second dialog box further includes a name field to display a name of a corresponding property and a default field to display a default configuration value associated with the corresponding property.

10. (Previously presented) The system of claim 9, wherein the second dialog box further includes a data type field to display the data type associated with corresponding property.

11. (Currently amended) A method comprising:

~~storing binaries and configuration data associated with a plurality of server nodes~~ within a data object, ~~the data object to store~~ a hierarchical representation of configuration data associated with ~~the server nodes~~ management of a plurality of server nodes organized as instances, each instance including at least one server process, the data object accessible by all of the servers and the hierarchy having a root, a first subset of nodes branching from the root containing configuration data associated with all servers in nodes branching from the root, a second subset of nodes branching from the root containing configuration data specific to all servers of a first instance and a third subset of nodes branching from the root of the second subset, the third subset of nodes containing configuration data specific to a first server of the first instance ~~and a plurality of nodes branching from the root;~~

providing one or more property sheets at one or more of the nodes, each of the property sheets including a plurality of configuration parameters associated with the server nodes, each parameter associated with a name, a default configuration value and optionally a custom configuration value;

pairing the default configuration value ~~to the~~ with an updated custom configuration value for a configuration parameter represented in the data object in response to the user specifying a custom parameter value ~~in place of a default parameter value~~ wherein both the default configuration value and the custom configuration value are preserved by the property data sheet; and

~~updating the configuration data of one of the server nodes upon receiving a parameter update request and responsively entering, in the data object, the custom configuration value stored in the property sheet associated with the updated server node in place of the default configuration value~~

determining whether to invalidate the configuration data stored on another of the server nodes based on the location of the updated configuration parameters within the hierarchy, wherein updated configuration parameters within the first subset of nodes invalidates configuration data of all servers, updated configuration parameters within the second subset of

nodes invalidates configuration data of all servers of the first instance and updated configuration parameters within the third subset of nodes invalidates configuration data of only the first server.

12. (Previously presented) The method as in claim 11 further comprising:

storing the data object, configuration data, binaries and property sheets within a central database, the central database accessible by the server nodes and a first node of the data object containing global configuration data associated with the plurality of server nodes and a second node of the data object containing configuration data specific to a one of the plurality of server nodes.

13. (Previously presented) The method of claim 11, wherein specifying a custom parameter value in place of a default parameter value in a property sheet, comprises:

opening the property sheet in a property sheet graphical user interface, the graphical user interface comprising a first column for storing parameter names, a second column for storing a current parameter value and a third column for storing an indication as to whether the current parameter value is a custom value or a default value;

selecting the indication in the third column;

responsively generating a data entry window having a custom field for entering a custom value; and

receiving user entry of a custom value in the custom field.

14. (Original) The method as in claim 11 wherein the server nodes are Java server nodes supporting the Java 2 Enterprise Edition (“J2EE”) standard and wherein the property sheet parameters comprise J2EE parameters.

15. (Currently amended) A method for updating configuration settings for a plurality of server nodes organized as instances, each instance including at least one server process, comprising:

storing, ~~binaries and configuration data associated with a plurality of server nodes within a data object, the data object to store~~ a hierarchical representation of configuration data associated with management of the server nodes, the data object accessible by all of the servers and the hierarchy having a root, a first subset of nodes branching from the root containing configuration data associated with all servers in nodes branching from the root, a second subset of nodes branching from the root containing configuration data specific to all servers of a first instance and a third subset of nodes branching from the root of the second subset, the third subset of nodes containing configuration data specific to a first server of the first instance ~~and a plurality of nodes branching from the root;~~

providing one or more property sheets at one or more of the nodes, each of the property sheets ~~including a plurality of configuration parameters associated with the server nodes,~~ each parameter associated with a name, a default configuration value and optionally a custom configuration value;

pairing the default configuration value ~~to the~~ with an updated custom configuration value for a configuration parameter represented in the data object in response to the user specifying a custom parameter value in place of a default parameter value wherein both the default configuration value and the updated custom configuration value are preserved by the property data sheet;

~~updating the configuration data of one of the server nodes upon receiving a parameter update request and responsively entering, in the data object, the custom configuration parameter value stored in the property sheet associated with the updated server node in place of the default configuration parameter value;~~

communicating an indication of the configuration parameter update to one or more other server nodes;

~~identifying in the data object the updated configuration parameters and~~ determining if the configuration data stored on the other server nodes is out-of-date based on the location of the updated configuration parameters within the hierarchy, wherein updated configuration parameters within the first subset of nodes renders all servers out-of-date, updated configuration parameters within the second subset of nodes renders all servers of the first instance out-of-date and updated configuration parameters within the third subset of nodes renders only the first server out-of-date; and

downloading the updated configuration data from the central database to the other server nodes if the configuration data stored on the other server nodes is out-of-date.

16. (Previously presented) The method as in claim 15 further comprising:  
acquiring a lock on the configuration parameters stored within the property sheet prior to updating the configuration parameters at the first server node.

17. (Previously presented) The method as in claim 16 further comprising:  
releasing the lock on the configuration parameters after the configuration data has been updated at the central database and/or at each of the other server nodes.

18. (Previously presented) A system comprising:  
server node means communicatively coupled on a network, the server node means to serve applications over the network to a plurality of clients;  
hierarchical data object means to store a hierarchical representation of configuration data associated with the server nodes, the hierarchical data object means having a root and a plurality of nodes branching from the root; and

property sheet means logically positioned at one of the nodes of the data object, the property sheet means including a plurality of property names, wherein each respective property name included in the property sheet means is associated with a default configuration value and, optionally, a custom configuration value to pair the default configuration value with the custom configuration value for a configuration parameter represented in the hierarchical data object means.

19. (Previously presented) The system as in claim 18 wherein the hierarchical data object means is stored within a central database accessible by each of the server nodes and a first node of the hierarchical data object means contains global configuration data associated with the plurality of server nodes and a second node of the hierarchical data object means contains configuration data specific to a one of the plurality of server nodes. .

20. (Previously presented) The system as in claim 18 wherein, once the default configuration value has been modified, the default configuration value is stored independently with respect to the custom parameters in the property sheet means.

21. (Previously presented) The property sheet means of claim 20, wherein the custom configuration values associated with the property sheet means are not user-modifiable via the user interface.

22. (Original) The property sheet system of claim 18, wherein the property sheet means is associated with a particular component or a set of components contained within the server node means.